MatInd and MatInspector: new fast and versatile tools for detection of consensus matches in nucleotide sequence data

Kerstin Quandt[1], Kornelie Frech[1], Holger Karas[2], Edgar Wingender[2] and Thomas Werner[1,*]

[1]Institut für Säugetiergenetik, GSF-Forschungszentrum für Umwelt und Gesundheit GmbH, Ingolstädter Landstraße 1, D-85758 Neuherberg, Germany and [2]Abteilung Genetik, Gesellschaft für Biotechnologische Forschung mbH, Mascheroder Weg 1, D-38124 Braunschweig, Germany

Received August 11, 1995; Revised and Accepted October 26, 1995

ABSTRACT

The identification of potential regulatory motifs in new sequence data is increasingly important for experimental design. Those motifs are commonly located by matches to IUPAC strings derived from consensus sequences. Although this method is simple and widely used, a major drawback of IUPAC strings is that they necessarily remove much of the information originally present in the set of sequences. Nucleotide distribution matrices retain most of the information and are thus better suited to evaluate new potential sites. However, sufficiently large libraries of pre-compiled matrices are a prerequisite for practical application of any matrix-based approach and are just beginning to emerge. Here we present a set of tools for molecular biologists that allows generation of new matrices and detection of potential sequence matches by automatic searches with a library of pre-compiled matrices. We also supply a large library (>200) of transcription factor binding site matrices that has been compiled on the basis of published matrices as well as entries from the TRANSFAC database, with emphasis on sequences with experimentally verified binding capacity. Our search method includes position weighting of the matrices based on the information content of individual positions and calculates a relative matrix similarity. We show several examples suggesting that this matrix similarity is useful in estimating the functional potential of matrix matches and thus provides a valuable basis for designing appropriate experiments.

INTRODUCTION

The location of nucleotide patterns is one of the most common tasks in sequence data analysis. Most commonly used sequence analysis software packages contain programs that are capable of rapidly finding nucleotide patterns, usually defined as IUPAC (1) coded strings (e.g. FindPatterns in the GCG package, Quest in the IG suite and Signal Scan; 2,3). They use a string such as SCAAK to represent all possible combinations GCAAG, CCAAG, GCAAT and CCAAT and to find every matching string. IUPAC-based search programs are very fast and do not require any input information other than the IUPAC code to find the appropriate pattern. However, the definition of the IUPAC code is to some extent arbitrary and is an inherent shortcoming of these algorithms.

Several methods have been published which attempt to locate consensus matches with more sophisticated algorithms than IUPAC searches. The salient feature distinguishing those methods from simple IUPAC searches is the use of all sequence information of the consensus sequences, rather than curtailing the information to an arbitrarily defined IUPAC code. Some of these methods require much more input data than for definition of a IUPAC code, in the form of surrounding sequences (4) or large training sets (5), in order to generate a consensus suitable for the location of matches in other sequences. This and the slower performance severely limit the applicability of these methods.

Nucleotide distribution matrices are more precise representations of consensus patterns than IUPAC strings. They utilize most of the sequence information and are thus more powerful and accurate. Some methods (including 6,7) use such a matrix approach to locate consensus matches.

We also chose a matrix representation to create a large library of consensus patterns and to develop a new search algorithm for the detection of these patterns in DNA sequences. Here we describe a simple but powerful method (MatInd) to derive a matrix description of a consensus from the same short sequences on which the definition of a IUPAC code is based. A large library of pre-defined matrix descriptions for protein binding sites exists and has been tested for accuracy and suitability.

The second software tool (MatInspector) utilizes this library of matrix descriptions to locate matches in other sequences. MatInspector is almost as fast as a IUPAC search, but is shown to produce superior results. It assigns a quality rating to matches and thus allows quality-based filtering and selection of matches.

[*] To whom correspondence should be addressed

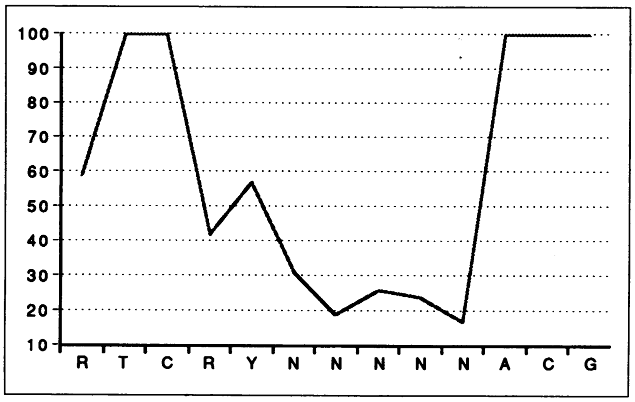

Figure 1. C_i vector for ABF1. The x-axis represents the binding matrix by an IUPAC code (solely for the purpose of presentation). The y-axis represents the C_i values ($0 \le C_i \le 100$).

MATERIAL AND METHODS

MatInd

The MatInd program constructs a description for a consensus (e.g. of a transcription factor binding site) which consists of a nucleotide distribution matrix and the conservation of each position within the matrix, represented by an array of values termed the consensus index vector (**C_i** vector).

MatInd expects input either in the form of a number of sequences representing the consensus (such as oligonucleotides used in binding assays; for an example see 8) or a nucleotide distribution matrix as present in the TRANSFAC database (9–11). In the case of sequences MatInd employs an alignment algorithm based on the method described by Frech *et al.* (4) and creates the nucleotide distribution matrix by counting the bases at each position of the alignment.

From this nucleotide distribution matrix (either pre-defined or created by the alignment procedure) the **C_i** vector is constructed by calculating the C_i value for each position i of the matrix:

$$C_i(i) = (100/\ln 5) \times [\sum_{b\in A,C,G,T,gap} P(i,b) \times \ln P(i,b) + \ln 5] \qquad (1)$$
$$0 \le C_i \le 100$$

where $P(i,b)$ is the relative frequency of nucleotide b at position i.

This **C_i** vector represents the conservation of the individual nucleotide positions in the matrix as numerical values and is used by the MatInspector program. The maximum C_i value of 100 is reached by a position with total conservation of one nucleotide, whereas the minimum value of 0 only occurs at a position with equal distribution of all four nucleotides and gaps (4).

The program MatInd also defines a core region within the matrix which is represented by the four consecutive nucleotide positions with the highest C_i sum. This core region of the matrix is used by MatInspector to pre-select potential matches.

For example, 22 sequences each containing an ABF1 binding site (Quandt *et al.*, in preparation) were aligned and yielded the following matrix and **C_i** vector (Table 1 and Fig. 1)

The IUPAC code for the ABF1 binding site is sometimes defined as RTCRYNNNNNACG (12) and sometimes as RTCRYYNNNNACG (13). In this case the last four positions within the matrix were determined as the core region because they yield the highest sum of C_i values for four consecutive nucleotide positions.

MatInspector

The program MatInspector uses the core, the nucleotide distribution matrix and the **C_i** vector created by MatInd to scan sequences of unlimited length for matches to the consensus matrix description.

The search starts with an optional pre-selection in which only matches to the core region are considered. This reduces the total number of matches and simultaneously accelerates performance of the program. A core similarity is calculated for each position of the sequence using equation **2**:

$$\text{core\_sim} = [\sum_{j=1}^{l} \text{score}(b,j)]/[\sum_{j=1}^{l} \text{max\_score}(j)] \qquad \textbf{(2)}$$
$$0 \le \text{core\_sim} \le 1$$

where l is the length of the core region, score(b,j) is the matrix value for base b at position j and max_score(j) is:

$$\max\{\text{score}(b,j)\}$$
$$b \in A, C, G, T$$

A matrix similarity is calculated according to equation **3** only if the core similarity reaches a user-defined threshold (this threshold can be set to zero, so that no core search is performed).

$$\text{mat\_sim} = [\sum_{j=1}^{n} C_i(j) \times \text{score}(b,j)]/[\sum_{j=1}^{n} C_i(j) \times \text{max\_score}(j)] \qquad \textbf{(3)}$$
$$0 \le \text{mat\_sim} \le 1$$

where $C_i(j)$ is the consensus index value of position j, n is the length of the consensus matrix, score(b,j) is the matrix value for base b at position j and max_score(j) is:

$$\max\{\text{score}(b,j)\}$$
$$b \in A, C, G, T$$

Table 1.

	Position												
	1	2	3	4	5	6	7	8	9	10	11	12	13
A	14	0	0	15	0	2	9	3	11	8	22	0	0
C	0	0	22	1	12	3	5	4	5	3	0	22	0
G	8	0	0	4	0	4	3	3	3	5	0	0	22
T	0	22	0	2	10	13	5	12	3	6	0	0	0
C_i	59	100	100	42	57	31	19	26	24	17	100	100	100

The core region is underlined

Matrix	pos.	1	2	3	4	5	6	7	8	9	10	11	12	13
	A	◆$\boxed{14}$	0	0	◆15	0	2	◆9	3	◆11	◆$\boxed{8}$	◆$\boxed{22}$	0	0
	C	0	0	◆$\boxed{22}$	1	◆$\boxed{12}$	3	$\boxed{5}$	4	$\boxed{5}$	3	0	◆$\boxed{22}$	0
	G	8	0	0	4	0	4	3	$\boxed{3}$	3	5	0	0	◆$\boxed{22}$
	T	0	◆$\boxed{22}$	0	$\boxed{2}$	10	◆$\boxed{13}$	5	◆12	3	6	0	0	0
	C_i:	**59**	**100**	**100**	**42**	**57**	**31**	**19**	**26**	**24**	**17**	**100**	**100**	**100**
Sequence		**A**	**T**	**C**	**T**	**C**	**T**	**C**	**G**	**C**	**A**	**A**	**C**	**G**

Numerator of mat_sim: Σ (score * C_i) of the sequence □ = 13426

score	14	22	22	2	12	13	5	3	5	8	22	22	22
	*	*	*	*	*	*	*	*	*	*	*	*	*
C_i	59	100	100	42	57	31	19	26	24	17	100	100	100

Denominator of mat_sim: Σ (maximum score * C_i) ◆ = 14426

score	14	22	22	15	12	13	9	12	11	8	22	22	22
(maximum)	*	*	*	*	*	*	*	*	*	*	*	*	*
C_i	59	100	100	42	57	31	19	26	24	17	100	100	100

Matrix similarity = 13426/14426 = 0.93

Figure 2. Determination of matrix similarity. The full nucleotide distribution matrix for 22 ABF1 binding sites is shown at the top of the figure. Diamonds indicate the most conserved nucleotides, while boxes represent the individual nucleotides present in the candidate sequence, which is shown below the C_i values of the matrix. The lower part of the figure details the calculation of the matrix similarity according to equation 3 (for further details see Material and Methods).

The matrix similarity reaches 1 only if the candidate sequence corresponds to the most conserved nucleotide at each position of the matrix. Multiplying each score by the C_i value emphasizes the fact that mismatches at less conserved positions are more easily tolerated than mismatches at highly conserved positions.

For example, scanning the sequence tccatctctcgcaacggcg for the ABF1 core region with a core similarity higher than 0.8 results in only position 13 (aacg) being selected. The matrix similarity for the sequence atctctcgcaacg surrounding the core region is 0.93 using the matrix for ABF1 defined above and equation **3**. Figure 2 demonstrates in full detail how this value is obtained.

Scanning the sequence aGGactataaacg, which has two mismatches with the ideal sequence (atcactataaacg) at conserved positions with high C_i values, yields a matrix similarity of only 0.695, whereas the sequence atcacAGtaaacg, with two mismatches at less conserved positions, yields 0.969.

Positions within the sequence to be analyzed that yield a matrix similarity above a user-defined threshold are written to an output file, together with the matrix similarity, the sequence matching the consensus and the name of the matrix, indicating the transcription factor bound.

Although a matrix description principally requires no more data than definition of a IUPAC string, individual sequences, or at least the nucleotide distribution matrix, must be known. We included a simple IUPAC search algorithm in MatInspector to allow use of pre-defined IUPAC strings where no sequences are available. In addition, this feature allows direct comparisons between results from IUPAC and matrix searches.

Matrix compilation from sequence data

TRANSFAC sites for matrices were automatically extracted from the database, including adjacent sequences from the EMBL data library. Both databases are generally linked to each other (11). Wherever possible sites were sorted according to the reliability of the experimental evidence. We used quality levels corresponding to unassigned binding sites (lowest quality), *bona fide* sequences, partially characterized binding sites (competition assays), binding sites confirmed by antibody supershifts, binding sites of purified proteins and binding sites with proven transcriptional activity (highest quality). Subsequently sequences were selected according to a pre-defined core, which included strand conversion if necessary. Sequence length was set to a defined site length (e.g. 7 nt for AP-1) with an additional two flanking positions on either side. The inclusion of a few additional nucleotides does not change the predictive value of the matrix as long as the matrix is not shorter than the actual binding site. These sequence elements were used to obtain alignments and nucleotide distribution matrices from MatInd, which also calculated the C_i values. We checked a 50 bp region around all sequence elements included in the matrices for higher scoring alternative positions and re-calculated the matrix wherever necessary.

Usage and availability of the programs and matrix library

Input for MatInd for construction of a matrix description is accepted as an IG formated file containing a number of binding sites or as a nucleotide distribution matrix as present in the TRANSFAC database. MatInspector is able to scan sequences of either the IG, GCG or EMBL format. The TRANSFAC library of pre-defined matrices for a variety of transcription factor binding sites is also available. This library includes explanations for those matrices that gave matches above the user-defined threshold and whose identifiers thus appear in the output. Information about the transcription factors connected to these matrices can also be retrieved. Another option to be implemented in the near future will enable the user to scan relevant information from the TRANSFAC database on the basis of a MatInspector output. All these data include relevant references.

The programs MatInd and MatInspector are written in ANSI C and are available for all UNIX computers (source code). PC and Macintosh versions (executables) are also available.

The programs MatInd and MatInspector, the matrix library and the TRANSFAC retrieval module can be obtained via anonymous ftp from ftp.gbf-braunschweig.de (193.175.244.2) or from ariane.gsf.de (146.107.21.33). The TRANSFAC database is available in ASCII flat file format at several sites (e.g. ftp.gbf-braunschweig.de or ftp.ebi.ac.uk).

RESULTS

Matrix library compiled by MatInd

The program MatInd has been used to compile a library of matrix descriptions for a variety of protein binding sites within nucleic acid sequences (Table 2). The library encompasses 214 matrices based on a total of 5701 individual sequences. Each matrix has thus been deduced from an average of 26 sequences, ranging from four or five sequences (two and eight matrices, respectively) up to 108 sequences. The final matrix library comprises four kinds of matrices.

Table 2. Summary on the nucleotide distribution matrices

	Group 1	Group 2	Group 3	Group 4	Total
Vertebrates	83	34	22	14[a]	153
Insects	12	16	1	–	29
Plants	2	2	1	–	5
Fungi	3	17	–	7[b]	27
Total	100	69	24	21	214

[a]This collection also comprises retroviral elements.

[b]Also containing one prokaryotic element (CRP).

Group 1. About 100 matrices taken from the literature are deduced from investigations of base preferences in the binding sites of defined transcription factors by random selection procedures.

Group 2. Also taken from the literature are matrices from compiled genomic (68 matrices) or artificial (1 matrix) binding sites.

Group 1 and 2 matrices are already part of the TRANSFAC database and have been assigned to one of the four major biological categories of the most investigated organisms: vertebrates, insects, plants and fungi. This assignment is reflected in the first character of the matrix identifier of TRANSFAC (e.g. V$E2F_01, I$HSF_01). These matrices are used by the MatInspector program without any further processing.

Group 3. We generated a number of matrices (24 matrices) from the genomic binding sites as compiled in the TRANSFAC database; these matrices were compiled by MatInd.

Group 4. Twenty one matrices were deduced from consensus descriptions previously generated by the ConsInd program (4).

The quality of a matrix definition is estimated by a value for random expectation (RE). This is defined as the number of matches with high matrix similarity (in our experience a matrix similarity ≥0.85) expected in a random sequence of 1000 bp. This RE value is calculated by multiplying the probabilities of occurrence for each C_i value of a matrix. A C_i value of 100 corresponds to a probability of 1/4, because only one single base is allowed at this position, and a C_i of 13.9 corresponds to a probability of one (all four bases may occur; Frech *et al.*, submitted). C_i values also represent the relative quantities of nucleotides present and, thereby, provide the most accurate basis for the estimation of random occurrence. The RE value is supplied for each matrix in the library.

To assess the quality of our RE we scanned 30 randomly generated sequences of 10 000 bp length for matches to three matrices with a matrix similarity ≥0.85 (Table 3).

Table 3.

Matrix	Random expectation for 10 000 bp	Mean observed
ADR1	138.4	156.5 ± 21.6
MATA1	5.4	7.5 ± 2.5
NIT1	25.3	37.6 ± 7.3

MatInspector results with pre-defined matrices

In order to test the capabilities of our matrix method we analyzed 4 600 000 bp of yeast genomic sequence (complete chromosomes II, III, VIII and XI) with a matrix derived from 22 binding sites for the yeast transcription factor ABF1. We chose this set-up as one of our test cases because a wealth of data about ABF1 binding sites is available which can be used to verify the results. Moreover, several IUPAC strings for the ABF1 binding site have been published which have been shown to provide a relatively precise description of ABF1 binding sites. Thus we were able to compare our method directly with well-defined IUPAC strings. As expected, we found that MatInspector detected all ABF1 sites matching the IUPAC string (cut-off matrix similarity ≥0.875). Additional sites were also detected.

The RTCRYNNNNNACG IUPAC string located 358 potential ABF1 sites in the 4 600 000 bp and the RTCRYYNNNNACG IUPAC string 228. We found 1171 matches to our ABF1 matrix with a similarity of ≥0.875. These include six ABF1 sites for which ABF1 binding had been experimentally verified (SCENOC, chr. VIII; COX6, chr. VIII; TRP3, chr. XI; PGK, chr. III; SCBAF1, chr XI; MATa, chr. III).

Table 4 shows a collection of experimentally verified ABF1 binding sites which include 11 ABF1 sites that have been shown to function in transcriptional regulation (only these sites will be referred to as functional). Note that all binding sites (except one, 0.888), including all three ABF1 sites not matching the IUPAC strings as well as all functional sites, have a matrix similarity ≥0.920.

The ABF1 binding site SCPK01 has been shown to bind ABF1 (14). Both IUPAC strings RTCRYYNNNNACG and RTCRYNNNNNACG fail to find this site, since position 4 (T) does not match the R (A or G) in the IUPAC strings. However, MatInspector located this binding site with a matrix similarity of 0.930. In summary, the results shown in Table 4 suggest a correlation between binding capacity and/or functionality and matrix similarity and consequently the three non-IUPAC ABF1 binding sites should be rated among biologically functional binding sites. However, this can only be conclusively established by experimental verification.

Table 4. Correlation of matrix similarity with functionality of ABF1 binding sites

Sequence name	Core similarity	Matrix name	Matrix similarity	Known functional	Sequence
SCCOXCH2	1.000	ABF1	0.962	+	atcattcccAACG
SCUBCOX8_INV	1.000	ABF1	0.921		gtcacgtggAACG
SCANB1RE_1	0.959	ABF1	0.950		atcatattcGACG
SCANB1RE_2	0.932	ABF1	0.888		gtcgtctcaCACG
SCMAT4	0.973	ABF1	0.950		atcataaaaTACG
SCMAT3_INV	0.973	ABF1	0.944		atcgccataTACG
M28606_INV	1.000	ABF1	0.970		atcattgcaAACG
SCPLASM[a]	1.000	ABF1	0.933		atctttgttAACG
SCHIS3G_DED1	0.973	ABF1	0.985	+	atcattctaTACG
SCHIS3G_DED2	1.000	ABF1	0.949	+	gtcattctgAACG
PGK	0.959	ABF1	0.951		atcacgagcGACG
SCPHO5_INV	0.959	ABF1	0.927		atcgttaatGACG
SCS33AA_INV	0.959	ABF1	0.967	+	gtcactctaGACG
SCTMC1A	0.973	ABF1	0.939	+	atcgttttgTACG
SCRGL2	0.932	ABF1	0.955	+	atcacgtcaCACG
SCBTUB_1_INV	0.932	ABF1	0.962	+	gtcactgtaCACG
SCBTUB_2	0.973	ABF1	0.954	+	gtcacgataTACG
SCBAF1[a]	1.000	ABF1	0.924		atccccattAACG
SCRPO31	0.973	ABF1	0.998		atcactataTACG
SCRPC40_INV	1.000	ABF1	0.975		gtcactataAACG
SCPK01[a]	1.000	ABF1	0.930		atctctcgcAACG
SCENOC	0.959	ABF1	0.945	+	gtcactaacGACG
ARS120_INV	0.932	ABF1	0.973		atcattatgCACG
COX6	0.973	ABF1	0.946	+	atcgctccaTACG
MATa	1.000	ABF1	0.968		atcattgaaAACG
TRP3	0.959	ABF1	0.962	+	atcactgacGACG

[a]Sites missed by both IUPAC strings.

+, ABF1 binding site functional in transcriptional regulation.

We found 360 matches with a matrix similarity of ≥0.92 in the 4 600 000 bp analyzed. Only 272 of these correspond to the IUPAC string RTCRYNNNNNACG. The data shown in Table 4 suggest that at least some of the remaining 88 non-IUPAC sites are probably binding sites for ABF1, since all known binding sites scored ≥0.92. Since there is insufficient data available, the number of false positives cannot be assessed. However, within the experimentally characterized 600 bp of the ENO2 sequence (12) the functional ABF1 site was the only match.

Similar results were obtained in three other examples. Binding to AP-1 sites (the most common IUPAC string, TGASTCA) was tested by Risse *et al.* (15) by analyzing binding capacities of oligonucleotides with consensus AP-1 binding sites mutated at single nucleotide positions. Our AP-1 binding site matrix derived from the binding sequences correctly separated 89% of binding and non-binding sequences (<25% of wild-type binding). The relaxed IUPAC string TKMSTCA allows correct identification of the same number of sequences as our matrix. However, TKMSTCA also matches sequences such as TTCCTCA, which do not resemble any known AP-1 binding site. This sequence is clearly separated from binding sites by a matrix similarity of only 0.764 (all binding sequences score ≥0.90 and even all non-binding sequences ≥0.813).

Further evidence of the suitability of the matrix score for detection of biological functionality was found in an example from the HIV genome, which contains a number of potential glucocorticoid elements (GRE), two of which are located in the LTR and the *vif* gene, respectively. Soudeyns *et al.* (16) found the *vif* GRE to be functional even in another context, while the LTR GRE was non-functional in its natural context but could be made functional in a different context. Both sites match the IUPAC string TGTYCT, which corresponds to the more conserved 3′ half-site of the GRE. MatInspector weighted both GRE sites correctly with respect to their relative functionality (matrix similarities: functional GRE in *vif* 0.826, partially functional LTR site 0.742).

Another example demonstrating the flexibility of the matrix similarity approach is the variant NF-Y binding site reported for the minute virus P4 promoter by Gu *et al.* (17). The authors stated that no such site could be found among >500 vertebrate promoters in the Eukaryotic Promoter Database (18) nor was such a sequence included in our matrix definition. Nevertheless, MatInspector located the experimentally verified site with a matrix similarity of 0.827 as the only match in the promoter region (200 bp) analyzed by Gu *et al.* (17).

The examples presented clearly show the superiority of the matrix search over simple IUPAC string scans. This has also been observed using matrices of Group 3 (deduced from TRANSFAC sites). No more than ~50% of the experimentally proven sites were detected (in most cases 30% or less) when analyzing all elements used to generate a matrix with the corresponding IUPAC string. In contrast, even with a restrictive threshold of 0.85, MatInspector recognized on average 88% (between 71 and 100%) of those sequences from which the matrices were deduced. The threshold allowing 90% of all sites to be found is between 0.77 and 0.93 for this group of matrices.

DISCUSSION

We have developed a set of software tools (MatInd and MatInspector) combined with a large library of pre-defined descriptions that allow fast scanning of sequence data for consensus motifs similar to IUPAC searches. Since our method is more versatile than IUPAC searches and assigns a quality to matches it should be very useful in pre-selecting potential regulatory sites for experimental studies.

Our method requires no more input data than the definition of a IUPAC string, but matrices defined by MatInd are more stable than IUPAC strings. Assignment of a IUPAC code is usually carried out by arbitrary definition of a representative majority for one or more nucleotides at each alignment position (formalized to some extent by Cavener; 19). As a direct consequence of this procedure, addition of a single sequence may change the resulting IUPAC string, while addition of a single sequence to a matrix has little effect on the matrix. IUPAC search algorithms may allow mismatches, but cannot rate them with respect to the position within the consensus.

MatInspector multiplies each score by the C_i value at individual positions, thus introducing an efficient and sensitive position weighting. Thus mismatches at less conserved positions are more easily tolerated than mismatches at highly conserved positions. As has been shown in the results, this allows detection of functional matches that deviate from the (IUPAC) consensus. Another advantage of the $\boldsymbol{C}_i$ vector compared with a IUPAC string is evident from the five Ns in the middle of the ABF1 binding site (Fig. 1). The matrix is sensitive to sequence conservation at all positions, thus matching the real situation in a better way. While the IUPAC algorithm does not distinguish between individual positions, the $\boldsymbol{C}_i$ vector indicates differences in the conservation of the positions: at position 8 all four bases occur, but T is more frequent than the others, leading to a higher C_i value.

We introduced an additional selection step by defining an arbitrary core of the four best conserved consecutive nucleotides. This represents a further emphasis on matrix matches that preserve this core. This proved to be useful in detection of protein binding sites, which have a strict requirement for conservation of a core sequence contacting the protein. However, this is an optional feature at the discretion of the user.

The matrix similarity assigns a quality value to each match in the scanned sequence, whereas a IUPAC string is restricted to a match/no match decision. We show this for two relatively long matrices with internal spacers (ABF1 and GRE), as well as for a highly conserved short matrix (AP-1). A total of 56 binding sites were analyzed with these three matrices. As suggested by the results shown in Table 4 (ABF1 example), matrix similarity appears to be correlated with biological functionality. The correlation of matrix similarity with biological functionality may be more general and not restricted to a particular class of matrices. Because the initial data available for construction of a matrix are highly heterogeneous in quality, it is necessary to estimate the reliability and specificity of a matrix. The RE value is a very simple approximation. However, it has proven its predictive value in several applications with real sequences, in addition to the random examples shown in this paper, and thus appears sufficient for this purpose. This should add to the practical applicability of our matrix library. The REs for all matrices in the library are included with the matrix identifier in order to facilitate matrix selection by the user. However, these values only provide additional information and are not involved in matrix similarity calculations.

The programs Consensus and FitConsensus in the GCG package (6) also define a matrix and are able to search sequences with this definition. Pre-aligned sequences are expected as input for the Consensus program. MatInd can either align a given set of sequences or even take matrix definitions as input if no sequences are available (which is the case in all matrices from Group 1), which adds to the usability of our approach. FitConsensus assigns a quality to a match in a sequence which is an absolute value. We chose to calculate the matrix similarity as a relative value in order to facilitate comparison with an ideal sequence. We also supply a library of pre-compiled matrices which can be tested against any number of sequences in only one run of the MatInspector program. Although FitConsensus has already shown the superiority of the matrix approach over IUPAC searches, we think that we have carried the idea further towards everyday applicability by incorporating the features discussed above.

MatInspector was designed to find matches within sequences as fast as possible, which necessitated the exclusion of alignment. However, MatInd carries out alignment and thus allows identification of subsets of binding sites according to spacer length. Thus MatInspector in principle is able to find binding sites with variable spacers provided that sufficient sequences for the generation of multiple matrices are available. Other methods are already available that allow more detailed assessment of the quality of a potential consensus match (e.g. ConsInspector; 4) not subject to the above restriction. However, in this case speed has to be traded for quality and the collection of consensuses accessible by the more advanced methods is much smaller than with the matrix library.

MatInspector greatly reduces the risk of missing functional sites (false negatives). Nevertheless, MatInspector may find false positives by scanning large genomic sequences. The sensitivity of the method remains to be established by comparison with sufficient experimental data once they become available, allowing assessment of false positive matches. This problem has to be overcome by considering the sequence context and thus the natural situation more precisely.

We would like to present our matrix search method as a versatile and general method for rapid detection of consensus matches with several advantages over IUPAC string searches.

ACKNOWLEDGEMENTS

The assistance of Ulrich Rückert in program development is gratefully acknowledged. This work was supported by BMBF ring funding project GENUS 413-4001-01 IB 306 D (Förderschwerpunkt Bioinformatik). This paper is dedicated to Prof. Dr J.Klein on the occasion of his 60th birthday

REFERENCES

1 Cornish-Bowden,A. (1985) *Nucleic Acids Res.*, **13**, 3021–3030.

2 Prestridge,D.S. (1991) *Comp. Appl. Biosci.*, **7**, 203–206.

3 Prestridge,D.S. and Stormo,G. (1993) *Comp. Appl. Biosci.*, **9**, 113–115.

4 Frech,K., Herrmann,G. and Werner,T. (1993) *Nucleic Acids Res.*, **21**, 1655–1664.

5 O'Neill,M.C. (1991) *Nucleic Acids Res.*, **19**, 313–318.

6 Staden,R. (1984) *Nucleic Acids Res.*, **12**, 505–519.

7 Hertz,G.Z., Hartzell,G.W. and Stormo,G.D. (1990) *Comp. Appl. Biosci.*, **6**, 81–92.

8 Liaw,P.C.Y. and Brandl,C.J. (1994) *Yeast*, **10**, 771–787.

9 Wingender,E. (1988) *Nucleic Acids Res.*, **16**, 1879–1902.

10 Wingender,E. (1994) *J. Biotechnol.*, **35**, 273–280.

11 Knüppel,R., Dietze,P., Lehnberg,W., Frech,K. and Wingender,E. (1994*) J. Comp. Biol.*, **1**, 191–198.

12 Brindle,P.K., Holland,J.P., Willett,C.E., Innis,M.A. and Holland,M.J. (1990) *Mol. Cell. Biol.*, **10**, 4872–4885.

13 Dhawale,S.S. and Lane,A.C. (1993) *Nucleic Acids Res.*, **21**, 5537–5546.

14 Chambers,A., Stanway,C., Tsang,J.S.H., Henry,Y., Kingsman,A.J. and Kingsman,S.M. (1990) *Nucleic Acids Res.*, **18**, 5393–5399.

15 Risse,G., Jooss,K., Neuberg,M., Brüller,H.-J. and Müller,R. (1989) *EMBO J.*, 8, 3825–3832.

16 Soudeyns,H., Geleziunas,R., Shyamala,G., Hiscott,J. and Wainberg,M.A. (1993) *Virology*, **194**, 758–768.

17 Gu,Z., Plaza,S., Perros,M., Cziepluch,C., Rommelaere,J. and Cornelis,J.J. (1995*) J. Virol.*, **69**, 239–246.

18 Bucher,P. (1990) *J. Mol. Biol.*, **212**, 563–578.

19 Cavener,D.R. (1987) *Nucleic Acids Res.*, **15**, 1353–1361.